DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD		TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
)	
	•	111

EX

ED SY LB

\$	CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC	RRRRRRRR RRRRRRRR RR RR RR RR RR RR RRRRRR		RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR	\$
		\$			

```
EDT$SCRCURS - conditionally position the cursor 16-Sep-1984 01:31:04 14-Sep-1984 12:24:23
                                                                                     VAX-11 Bliss-32 V4.0-742 Pa
DISK$VMSMASTER:[EDT.SRC]SCRCURS.BLI;1
       O %TITLE 'EDT$SCRCURS - conditionally position the cursor'
                                                                  ! Conditionally position the cursor
! File: SCRCURS.BLI Edit: JBS1012
0003
       O MCDULE EDTSSCRCURS (
                            IDENT = 'V04-000'
0004
0005
         BEGIN
0006
0007
8000
             COPYRIGHT (c) 1978, 1980, 1982, 1984 BY DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0009
       1 1.
0010
0011
              ALL RIGHTS RESERVED.
0012
       1 !* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
             ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
       1 !*
0014
       1 1 .
0015
0016
              COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
       1 !*
0017
         1 🛊
              OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0018
         l 🛊
              TRANSFERRED.
0019
         i 🛊
0020
         ! THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
       1 !*
              AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0021
0022
              CORPORATION.
       1 1 *
         1 🛊
0024
         1 🏚
              DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0025
              SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
         1 *
0026
          1 *
0027
         1 *
```

! FACILITY: EDT -- The DEC Standard Editor

ABSTRACT:

....

0028

0034

0035 0036

0037

0038 0039

0040 0041

0042

0044

0045 0046 0047

0048

10

11

18

22222222222333333333334

489012345557

This module puts out a cursor position sequence to a specified line and column, unless the cursor is already there.

.................

ENVIRONMENT: Runs at any access mode - AST reentrant AUTHOR: Bob Kushlis, CREATION DATE: September 8, 1979

MODIFIED BY:

```
1-001 - Original. DJS 12-Feb-1981. This module was created by extracting the routine EDT$$SC_POSCSIF from module SCREEN.

1-002 - Regularize headers and don't store into the parameter list. This change sayed two words on the PDP-11. JBS 13-Mar-1981
1-003 - Always position to the last column, to avoid autowrap problems.

JBS 15-Apr-1982

1-004 - Remove edit 1-003, since it doesn't help a VT100, and add some cursor positioning optimizations. for now, turn off the cursor positioning optimizations, until EDT passes the QA system. JBS 05-Oct-1982

1-005 - Turn the cursor positioning optimizations on, to see if EDT still passes the QA system. JBS 15-Oct-1982

1-006 - When positioning below the scrolling region don't use the relative cursor motion commands. JBS 15-Oct-1982
```

```
EDT$SCRCURS - conditionally position the cursor 16-Sep-1984 01:31:04 Declarations 14-Sep-1984 12:24:23
EDT$SCRCURS
V04-000
                                                                                                                 VAX-11 Bliss-32 V4.0-742 Page 3 DISK$VMSMASTER:[EDT.SRC]SCRCURS.BLI;1 (2)
                    0069
0070
0071
                           1 %SBTTL 'Declarations'
   1 ! TABLE OF CONTENTS:
                    0072
0073
                               REQUIRE 'EDTSRC: TRAROUNAM';
                            1 FORWARD ROUTINE
1 EDT$$SC_POSCSIF : NOVALUE;
                                 INCLUDE FILES:
                               REQUIRE 'EDTSRC:EDTREQ';
                                 MACROS:
                    0660
                                         NONE
                    0661
                    0662
0663
                                 EQUATED SYMBOLS:
                    0664
                                         NONE
                    0666
0667
0668
                                 OWN STORAGE:
                                         NONE
                    0669
0670
                                 EXTERNAL REFERENCES:
                    0671
                    0672
                           1 !
                                         In the routine
```

```
EDT$SCRCURS - conditionally position the cursor 16-Sep-1984 01:31:04 EDT$$SC_POSCSIF - conditionally position the c 14-Sep-1984 12:24:23
EDT$SCRCURS
                                                                                                                                VAX-11 Bliss-32 V4.0-742 Pa
DISK$VMSMASTER:[EDT.SRC]SCRCURS.BLI;1
V04-000
                                1 %SBTTL 'EDT$$SC_POSCSIF - conditionally position the cursor'
                       0674
0675
0676
0677
0678
    104
    105
                                   GLOBAL ROUTINE EDT$$SC_POSCSIF (
                                                                                                          ! Conditionally position the cursor ! Desired line
    106
                                        LĪNE,
                                         POS
    107
                                                                                                            Desired column
    108
                                         ) : NOVALUE =
    109
                       0680
    110
                       0681
0682
0683
    111
                                   ! FUNCTIONAL DESCRIPTION:
    112
    113
                                              This routine positions the cursor to [LINE, POS]. No output is
                       0684
0685
    114
                                              produced if the cursor is already positioned correctly.
    115
                       0686
    116
                                     FORMAL PARAMETERS:
    117
                       0687
    118
                       0688
                                       LINE
                                                                      Desired line
                       0689
    119
   0690
                                       POS
                                                                      Desired column
                       0691
                       0692
0693
                                      IMPLICIT INPUTS:
                                              EDT$$G_PRV_LN
EDT$$G_PRV_COL
EDT$$G_BOT_SCREG
EDT$$G_TI_DUMB
EDT$$G_TI_WID
                       0694
                       0695
                       0696
                       0697
                       0698
                       0699
                       0700
                                     IMPLICIT OUTPUTS:
                       0701
                       0702
                                              NONE
                       0703
                       0704
                                     ROUTINE VALUE:
                       0705
0706
0707
                                              NONE
                       0708
0709
                                     SIDE EFFECTS:
                       0710
    140
                                              May position the cursor.
    141
                       0711
   142
                       0712
0713
    144
                       0714
                                        BEGIN
    145
                       0715
                                        EXTERNAL ROUTINE
EDT$$SC_POSABS : NOVALUE
EDT$$STORE_FMTCH : NOVALUE;
EDT$$FMT_DCML : NOVALUE;
    146
                       0716
   147
                       0717
                                                                                                          ! Force a cursor position sequence to the output buffer
                       0718
                                                                                                            Store a character in the format buffer
   149
150
151
152
153
154
155
156
157
158
159
                       0719
                                                                                                         ! Place a decimal number in the format buffer
                       0720
                       0721
0722
0723
0724
0725
0726
0727
0728
0729
                                         EXTERNAL
                                              ENNAL
EDT$$G_BOT_SCREG,
EDT$$G_PRV_LN,
EDT$$G_PRV_COL,
EDT$$G_TI_TYP,
EDT$$G_TI_DUMB,
EDT$$G_TI_DUBS,
EDT$$G_TI_DID;
                                                                                                            Bottom of current scrolling region Previous line number
                                                                                                            Previous column number
Terminal type, either VT52 or VT100
                                                                                                          ! 1 = this is a dumb terminal driver
                                                                                                         ! Cursor position, for formatting purposes. ! Terminal width
```

VÕ4

; F

```
VAX-11 Bliss-32 V4.0-742 Pa
DISK$VMSMASTER:[EDT.SRC]SCRCURS.BLI;1
EDT$SCRCURS
                  EDT$SCRCURS - conditionally position the cursor 16-Sep-1984 01:31:04
                  EDT$$SC_POSCSIF - conditionally position the c 14-Sep-1984 12:24:23
V04-000
                  0730
0731
                                    SAVE_FMTLNPOS:
   161
   162
                               ASSERT (.LINE GEQ 0);
                          If we are already positioned correctly, do nothing.
   165
                  0735
                  0736
0737
0738
   167
   168
                                IF ((.LINE EQL .EDT$$G_PRV_LN) AND (.POS EQL .EDT$$G_PRV_COL)) THEN RETURN;
  169
170
                  0739
                  0740
   171
                  0741
                             Preserve EDT$$G_FMT_LNPOS, since the calls to EDT$$FMT_DCML will increment it,
  172
173
                  0742
0743
                             but those characters do not in fact move the cursor, since they are in a control sequence.
   174
                  0744
                               SAVE_FMTLNPOS = .EDT$$G_FMT_LNPOS:
   175
                  0745
                  0746
   176
                           ! If this is a dumb terminal driver, we are positioning below the scrolling
   177
                  0747
                             region, or coming from the right margin, issue an absolute cursor position.
   178
                  0748
                             A dumb terminal driver is one which does not permit its autowrap logic to be disabled.
   179
                  0749
   180
                  0750
   181
                  0751
                                IF (((.LINE + 1) GEQ .EDT$$G_BOT_SCREG) OR (.EDT$$G_PRV_COL EQL (.EDT$$G_TI_WID - 1)) OR .EDT$$G_TI_DUMB
   182
183
184
185
                 0752
0753
0754
                               THEN
                                    BEGIN
                                    EDT$$SC_POSABS (.LINE, .POS);
EDT$$G_FMT_LNPOS = .SAVE_FMTLNPOS;
                  0755
                 0756
0757
0758
   186
                                    RETURN:
   187
                                    END:
   188
189
                  0759
   190
191
192
193
194
195
                 0760
                           ! If the desired column is zero, but the current column is non-zero, issue a
                  0761
                             carriage return provided that the rows match, so that we will not be issuing
                 0762
0763
                             an absolute cursor position later.
                 0764
                               IF ((.POS EQL 0) AND (.EDT$$G PRV_COL NEQ U) AND !
((.LINE EQL .EDT$$G_PRV_LN) OR (.LINE EQL (.EDT$$G_PRV_LN + 1))))
                 0765
   0766
                  0767
                               THEN
                  0768
                                    BEGIN
                 0769
0770
                                    EDT$$STORE_FMTCH (ASC_K_CR);
                                    EDT$$G_PRV_COL = 0;
                  0771
                                    END:
                  0772
0773
                           ! If the column is correct but the row is incorrect, issue an appropriate
                  0774
                  0775
                           ! sequence.
                  0776
0777
                  0778
                                IF ((.POS EQL .EDT$$G_PRV_COL) AND (.LINE NEQ .EDT$$G_PRV_LN))
                  0779
                                THEN
                  0780
                                    BEGIN
                  0781
                        If we are moving down one, issue a line feed.
                  0782
0783
                  0784
                  0785
                                    IF (.LINE EQL (.EDT$$G_PRV_LN + 1))
                  0786
```

EDT VO4

```
EDT$SCRCURS
                                         EDT$SCRCURS - conditionally position the cursor 16-Sep-1984 01:31:04
                                                                                                                                                                                                                                    VAX-11 Bliss-32 V4.0-742 Particular Particul
V04-000
                                         EDT$$SC_POSCSIF - conditionally position the c 14-Sep-1984 12:24:23
      0788
                                                                                              EDT$$STORE_FMTCH (ASC_K_LF);
                                          0789
                                                                                              EDT$$G_PRV_LN = .EDT$$G_PRV_LN + 1;
                                          0790
                                                                                              END
                                          0791
                                                                                   ELSE
                                          0792
0793
                                                                   Otherwise issue an appropriate relative cursor motion command. Note that the VT52
                                          0794
                                                                   can only move by one.
                                          0795
                                          0796
                                          0797
                                                                                              IF ((.EDT$$G_TI_TYP EQL TERM_VT100) OR (ABS (.LINE - .EDT$$G_PRV_LN) EQL 1))
                                          0798
                                                                                              THEN
                                          0799
                                                                                                        BEGIN
                                          0800
                                                                                                        EDT$$STORE_FMTCH (ASC_K_ESC);
                                          0801
                                          0802
                                                                                                        IF (.EDT$$G_TI_TYP EQL TERM_VT100) THEN EDT$$STORE_FMTCH (%C'[');
                                          0803
                                          0804
                                                                                                        IF (.LINE LSS .EDT$$G_PRV_LN)
                                          0805
                                                                                                        THEN
                                          0806
                                                                                                                  BEGIN
                                          0807
                                          8080
                                                                                                                  IF ((.EDT$$G_PRV_LN - .LINE) NEQ 1) THEN EDT$$FMT_DCML (.EDT$$G_PRV_LN - .LINE);
                                          0809
                                                                                                                  EDT$$STORE_FMTCH (%C'A');
EDT$$G_PRV_LN = .LINE;
EDT$$G_PRV_COL = .POS;
                                          0810
                                          0811
                                         0812
0813
                                                                                                                  END
                                          0814
                                                                                                        ELSE
                                         0815
                                                                                                                  BEGIN
                                                                                                                  EDT$$FMT_DCML (.LINE - .EDT$$G_PRV_LN);
EDT$$STORE_FMTCH (%C'B');
                                         0816
                                         0817
                                         0818
                                                                                                                  EDT$$G_PRV_LN = .LINE;
                                         0819
                                                                                                                  EDT$$G_PRV_COL = .POS;
                                         0820
                                                                                                                  END:
                                          0821
                                          0822
                                                                                                        END:
                                         0823
                                         0824
                                                                                  END;
                                         0825
                                         0826
                                         0827
                                                                  Now check for moving left or right. Note again that the VT52 can only
                                         0828
                                                                   move by one.
                                         0829
                                         0830
                                         0831
                                                                         IF ((.POS NEQ .EDT$$G_PRV_COL) AND (.LINE EQL .EDT$$G_PRV_LN))
                                         0832
0833
                                                                         THEN
                                                                                   BEGIN
                                          0834
                                          0835
                                                              ! If we are just going back one, use BS.
                                         0836
0837
                                         0838
                                                                                   IF (.POS EQL (.EDT$$G_PRV_COL - 1))
                                          0839
                                                                                    THEN
                                          0840
                                                                                             BEGIN
                                          0841
                                                                                             EDT$$STORE_FMTCH (ASC_K_BS);
                                                                                             EDT$$G_PRV_COL = .EDT$$G_PRV__OL - 1;
```

* * 5

```
EDTSSCRCURS - conditionally position the cursor 16-Sep-1984 01:31:04
                                                                                                                                                                                                                                           VAX-11 Bliss-32 V4.0-742 Patricks Patrick Patr
EDT$SCRCURS
                                                                                                                                                                                                                                                                                                                                            Page
V04-000
                                          EDT$$SC_POSCSIF - conditionally position the c 14-Sep-1984 12:24:23
                                                                                     ELSE
       2757789012328288889012345678901
277778901232888888999329998901
377890123222222222222333
                                          0845
                                           0846
                                                                                                 IF ((.EDT$$G_TI_TYP EQL TERM_VT100) OR (ABS (.POS - .EDT$$G_PRV_COL) EQL 1))
                                           0847
                                                                                                THEN
                                          0848
                                                                                                           BEGIN
                                          0849
                                                                ! Otherwise, use a relative cursor motion command.
                                          0850
                                           0851
                                          0852
0853
                                                                                                          EDT$$STORE_FMTCH (ASC_K_ESC);
                                           0854
                                                                                                           IF (.EDT$$G_TI_TYP EQL TERM_VT100) THEN EDT$$STORE_FMTCH (%C'[');
                                           0855
                                          0856
                                                                                                           IF (.POS LSS .EDT$$G_PRV_COL)
                                                                                                           THEN
                                           0858
                                                                                                                      BEGIN
                                                                                                                     EDT$$FMT_DCML (.EDT$$G_PRV_COL - .POS);
EDT$$STORE_FMTCH (%C'D');
EDT$$G_PRV_COL = .POS;
                                           0859
                                          0860
                                          0861
                                          0862
                                                                                                                     END
                                           0863
                                                                                                          ELSE
                                          0864
                                                                                                                      BEGIN
                                          0865
                                          0366
                                                                                                                     IF ((.POS - .EDT$$G_PRV_COL) NEQ 1) THEN EDT$$FMT_DCML (.POS - .EDT$$G_PRV_COL);
                                          0867
                                          0868
                                                                                                                     EDT$$STORE_FMTCH (%C'C');
EDT$$G_PRV_COL = .POS;
                                          0869
                                          0870
                                                                                                                     END:
                                          0871
       302
                                          0872
                                                                                                           END:
       303
       304
                                          0874
                                                                                     END:
       305
                                          0875
       306
                                          0876
       307
                                          0877
                                                                    If the above logic does not succeed in getting the cursor positioned correctly,
       308
                                          0878
                                                                    do it using an absolute cursor position sequence.
                                          0879
       309
       310
                                          0880
       311
                                          0881
                                                                           IF ((.LINE NEQ .EDT$$G_PRV_LN) OR (.POS NEQ .EDT$$G_PRV_COL))
       312
313
                                          0882
                                                                                     EDT$$SC_POSABS (.LINE, .POS);
                                          0884
                                          0885
                                          0886
                                                                    Restore the format line position.
       317
318
                                          0887
                                          0888
                                                                           EDT$$G_FMT_LNPOS = .SAVE_FMTLNPOS;
                                          0889
                                                                           RETURN:
       320
                                          0890
                                                                                                                                                                                                 ! of routine EDT$$SC_POSCSIF
                                                                           END:
                                                                                                                                                                                                      .TITLE EDT$SCRCURS EDT$SCRCURS - conditionally positio
                                                                                                                                                                                                                                                           n the cursor
                                                                                                                                                                                                      .IDENT \V04-000\
                                                                                                                                                                                                                          EDT$$SC_POSABS, EDT$$STORE_FMTCH
EDT$$FMT_DCML, EDT$$G_BOT_SCREG
EDT$$G_PRV_LN, EDT$$G_PRV_COL
                                                                                                                                                                                                      .EXTRN
                                                                                                                                                                                                      .EXTRN
                                                                                                                                                                                                      .EXTRN
                                                                                                                                                                                                                          EDT$$GTI_TYP, EDT$$GTI_DUMB
                                                                                                                                                                                                      .EXTRN
```

EDT\$SCRCURS V04-000 M 13
EDT\$SCRCURS - conditionally position the cursor 16-Sep-1984 01:31:04 VAX-11 Bliss-32 V4.0-742 Page 8
EDT\$\$SC_POSCSIF - conditionally position the c 14-Sep-1984 12:24:23 DISK\$VMSMASTER:[EDT.SRC]SCRCURS.BLI;1 (3)

•		• •				•			
							.EXTRN .EXTRN	EDT\$\$G_FMT_LNPOS EDT\$\$G_TI_WID, EDT\$\$INTER_ERR	
							.PSECT	_EDT\$CODE,NOWRT, SHR, PIC,2	
			0	FFC	00000		.ENTRY	EDT\$\$SC_POSCSIF, Save R2,R3,R4,R5,R6,R7,R8,-;	0675
	5B 5A	0000000G	00	9E 9E 9E 9E	00002		MOVAB MOVAB	R9,R10,R11 EDT\$\$FMT DCML, R11 EDT\$\$G_TI_TYP, R10	ı I
	59 58	0000000nG	00 00 00 AC	9E 9E	00010		MOVAB MOVAB	EDTSSG_TT_TYP, R10 EDTSSSTORE_FMTCH, R9 EDTSSG_PRV_LN, R8 EDTSSG_PRV_COL, R7 LINE, R4	,
	57 54	00000000G	AC 07	D0 18	0001E 00025 00029		MOVAB Movl BGEQ	LINE, R4	0733
00000000G	00 68		00 54	FB D1	0002B 00032	15:	CALLS	#O, EDT\$\$INTER_ERR R4, EDT\$\$G_PRV_LN	0738
	67	08	07 AC	12 D1	00035 00037		BNEQ CMPL	POS, EDT\$\$G_PRV_COL	
			01	12	0003B 0003D		BNEQ RET	2\$	
0000000	56 55	00000000G 01	A4	00 9E	0003E 00045	25:	MOVL MOVAB	EDT\$\$G_FMT_LNPOS, SAVE_FMTLNPOS 1(R4), R5	0744
000000006	00		55 14	D1 18	00049		CMPL BGEQ SUBL3	RS, EDT\$\$G_BOT_SCREG	1 1
50 00000000G	00 50		01 67	C3	00052 0005A		CMPL	#1, EDT\$\$G_TI_WID, RO EDT\$\$G_PRV_COL, RO	
	06	00000000G	07 00 AC	13 E9 DD	0005D 0005F 00066	7¢.	BEQL BLBC PUSHL	3\$ EDT\$\$G_TI_DUMB, 4\$	0754
	53		010F	31	00069		BRW	POS 24 \$ POS, R3	0765
	,,	00	AC 1A 67	D0 12 D5	00070	40.	MOVL BNEQ TSTL	6\$ EDT\$\$G_PRV_COL	
	50		16	13 00	00074		BEQL MOVL	6\$ EDT\$\$G_PRV_LN, RO	0766
	5ŏ		68 54 07	D1 13	00079 00070		CMPL BEQL	R4, R0 5 \$:
	50		50	D6	0007E		INCL	RO ; R4, RO ;	
			54 07 00 01	12	00080 00083 00085	5 \$:	CMPL BNEQ PUSHL	6 \$ #13	0769
	69		01 67	FB D4	00087 0008A		CALLS CLRL	#1, EDT\$\$STORE_FMTCH ; EDT\$\$G_PRV_COL ;	0770
	67		67 53 60 54	D1 12	0008C 0008F	6\$:	CMPL BNEQ	R3, EDT\$\$G_PRV_COL	0778
	68		54 67	D1 13	00091 00094		CMPL BEQL	R4, EDT\$\$G_PRV_LN 14\$	
	52 50 50	01	68 A2 54	DO 9E	00096		MOVL MOVAB	EDT\$\$G_PRV_LN, R2; 1(R2), R0;	0785
	50	- ,	54 09	D1 12	00099 00090 000A0		CMPL BNEQ	R4, R0 7 \$	
	69		0A 01	DD FB	000A2		PUSHL Calls	#10 ; #1, EDT\$\$STORE_FMTCH ;	0788
			68 52	D6 11	900A7	_	INCL BRB	EDT\$\$G_PRV_LN ;	0789 0785
	02		6A OE	D1 13	000A9 000AB 000AE	7\$:	CMPL Begl	EDT\$\$G_TI_TYP, #2 ; 9\$	0797

EDT\$SCRCURS V04-000	EDT\$SCRCURS - cond EDT\$\$SC_POSCSIF	ditionally p - conditiona	osit ⁱ or lly pos	n the	N 13 cursor 16-Sep the c 14-Sep	0-1984 01:3 0-1984 12:2	1:04 VAX-11 Bliss-32 V4.0-742 4:23 DISK\$VMSMASTER:[EDT.SRC]SC	Page 9 RCURS.BLI;1 (3)
	50	54		52	C3 000B0 18 000B4	SUBL3	R2, R4, R0	;
		50 01		52 50 50 50 51 18 01	CE 000B6	BGEQ MNEGL	8\$° RO, RO RO, #1	
		O I		3F	12 000BC	CMPL BNEQ	RÓ, RO RO, W1 14\$ W27	. 0800
		69 02			DD 000BE 93: FB 000CO D1 000C3	PUSHL CALLS CMPL	W1, EDT\$\$STORE_FMTCH EDT\$\$G_TI_TYP, W2	; 0800 ; 0802
			5B	07 8F 21	12 00006 9A 00008	BNEQ MOVZBL	10\$ #91, -(SP)	:
		7E 69 50 50		21 68	FB 000CC D0 000CF 10\$	CALLS	#1. EDT\$\$STORE FMTCH	0804
				68 54 12 50 07	D1 000D2 18 000D5	CMPL BGEQ	EDT\$\$G_PRV_LN, RO R4, RO 12\$	
		55		50 07	D1 000D7	(MPI	RO, R5 11 \$	0808
	7E	50 6B 7E		54 01 8F	13 000DA C3 000DC FB 000E0 9A 000E3 11\$	SUBL3 CALLS	R4, R0, -(SP) #1, EDT\$\$FMT_DCML	
	75		41	0B	11 000E7	BRB	13\$: 0810
	7 E	54 68 7E 69 68 67 55	, 2	50 01	C3 000E9 12\$ FB 000ED	SUBL3 CALLS MOVZBL	RO, R4, -(SP) #1, EDT \$\$ FMT_DCML #66, -(SP)	0816
		69 68	42	8F 01 54	9A 000F0 FB 000F4 13\$ D0 000F7	CALLS	#1 ENTERCYODE EMTCH	0817
		67 55		5 <u>3</u>	DO 000FA DO 000FD 14\$	MOVL	R3, EDT\$\$G_PRV_COL	; 0811 ; 0812 ; 0831
		55		54 53 67 53 6A	D1 00100 13 00103	CMPL BEQL	R4, EDT\$\$G_PRV_LN R3, EDT\$\$G_PRV_COL EDT\$\$G_PRV_COL, R5 R3, R5 22\$ R4, EDT\$\$G_PRV_LN 22\$. 0031
		68		54 65	D1 00100 13 00103 D1 00105 12 00108 9E 0010A	CMPL BNEQ	R4. EDT\$\$G_PRV_LN	
		50 50	FF	A5 53	9E 0010A D1 0010E	MOVAB CMPL	-1(KJ), KU	0838
				09	12 00111	BNEQ	15 \$ #8	0841
		69		6/	DD 00113 FB 00115 D7 00118	PUSHL CALLS DECL BRB	#1, EDT\$\$STORE_FMTCH	0842
		02		53 6A	11 0011A D1 0011C 15\$:	CMDI	22\$ EDI\$\$G_TI_TYP, #2	; 0838 ; 0846
	50	53		6A 0E 55 03 50 50 40 1B	13 0011F C3 00121 18 00125 CE 00127 D1 0012A 16\$: 12 0012D DD 0012F 17\$: FB 00131 D1 00137	BEQL SUBL3	22\$ EDT\$\$G_TI_TYP, #2 17\$ R5, R3, R0 16\$	
		50 01		50 50	CE 00127	BGEQ MNEGL	RO, RO	
		O I		40 18	12 0012b	CMPL BNEQ PUSHL	RO, RO RO, W1 22\$ W27 W1, EDT\$\$STORE_FMTCH	0852
		69 02		01 6A	FB 00131	CALLS	#1, EDT\$\$STORE_FMTCH FDT\$\$G II IVP #2	0854
			58	07 8f 01	12 00137 9A 00139	CMPL BNEQ MOVZBL	EDT\$\$G_TI_TYP, "M2 18\$ M91, -(SP)	
		7E 69 55 55		67	9Ã 00139 FB 0013D D0 00140 18\$:	CALLS	#1. FDISSSIORE FMICH	0856
				53 00 53	D1 00143 18 00146	CMPL BGEQ_	EDT\$\$G_PRV_COL, R5 R3, R5 19\$ R3, R5, -(SP)	:
	7E	55 6B 7E		01	PB 0014L	BGEQ SUBL3 CALLS MOVZBL	19\$ R3, R5, -(SP) #1, EDT\$\$FMT_DCML #68, -(SP)	0859
		7E	44	8F	9A 0014F	MOVZBL	#68, -(SP)	: 0860

EDT\$SCRCURS V04-000	EDT\$SCRCURS - conditio EDT\$\$SC_POSCSIF - con	nally p ditiona	osition the	ne cu ion t	urso- 10 the c 10	3 14 5-Sep- 4-Sep-	1984 01:31 1984 12:24	: 04 : 23	VAX-11 Bliss-32 V4.0-742 PDISK\$VMSMASTER:[EDT.SRC]SCRCURS.BLI;1	Page 10 (3)
		50 50	01 A		00153 00155 00159 00150 0015E	19\$:	BRB MOVA B (MPL BEQL_	21\$ 1(R5) R3 R 20\$	δ ^{RO}	0866
	7E	53 6B 7E 69	43 81 0	C3 FE 9/	00162	20\$:	SUBL3 CALLS MOVZBL	R5, R #1, E #67,	3, -(SP) DT\$\$FMT_DCML -(SP) DT\$\$STORE_FMTCH	0868
		67 68	5: 5: 0:	5 D(5 D) 5 12	0016C		CALLS MOVL CMPL BNEQ	R3, E R4, E 23\$	DT\$\$G_PRV_COL DT\$\$G_PRV_LN	0861 0881
		67	5) 0) 5) 54	3 13 3 13 5 DI	00177 00179 0017B	23 \$: 24 \$:	CMPL BEQL PUSHL PUSHL	25 5 R3 R4	DT\$\$G_PRV_COL	0883
	0000000G 0000000G	00	5) 0) 5)	2 FE 5 D(0017D 00184 0018B	25\$:	CALLS MOVL RET	M2, E SAVE_	DT\$\$SC_POSABS FMTLNPOS, EDT\$\$G_FMT_LNPOS	0888 0890

; Routine Size: 396 bytes, Routine Base: _EDT\$CODE + 0000

0891 1 0892 1 !<BLF/PAGE> ; 321 ; 322

ED VO

PSECT SUMMARY

Name Bytes Attributes

_EDT\$CODE 396 NOVEC, NOWRT, RD, EXE, SHR, LCL, REL, CON, PIC, ALIGN(2)

Library Statistics

		- Symbols		Pages	Processing	
File	Total	Loaded	Percent	Mapped	Time	
_\$255\$DUA28:[EDT.SRC]EDT.L32;1 _\$255\$DUA28:[EDT.SRC]PSECTS.L32;1	377	6	1 50	40 7	00:00.2 00:00.1	

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD, INITIAL, OPTIMIZE)/NOTRACEBACK/LIS=LIS\$:SCRCURS/OBJ=OBJ\$:SCRCURS MSRC\$:SCRCURS.BLI/UPDATE=(ENH\$:SCRCURS

; Size: 396 code + 0 data bytes; Run Time: 00:20.2; Elapsed Time: 00:23.6; Lines/CPU Min: 2653; Lexemes/CPU-Min: 7929; Memory Used: 138 pages

; Compilation Complete

0138 AH-BT13A-SE VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION CONFIDENTIAL AND PROPRIETARY

